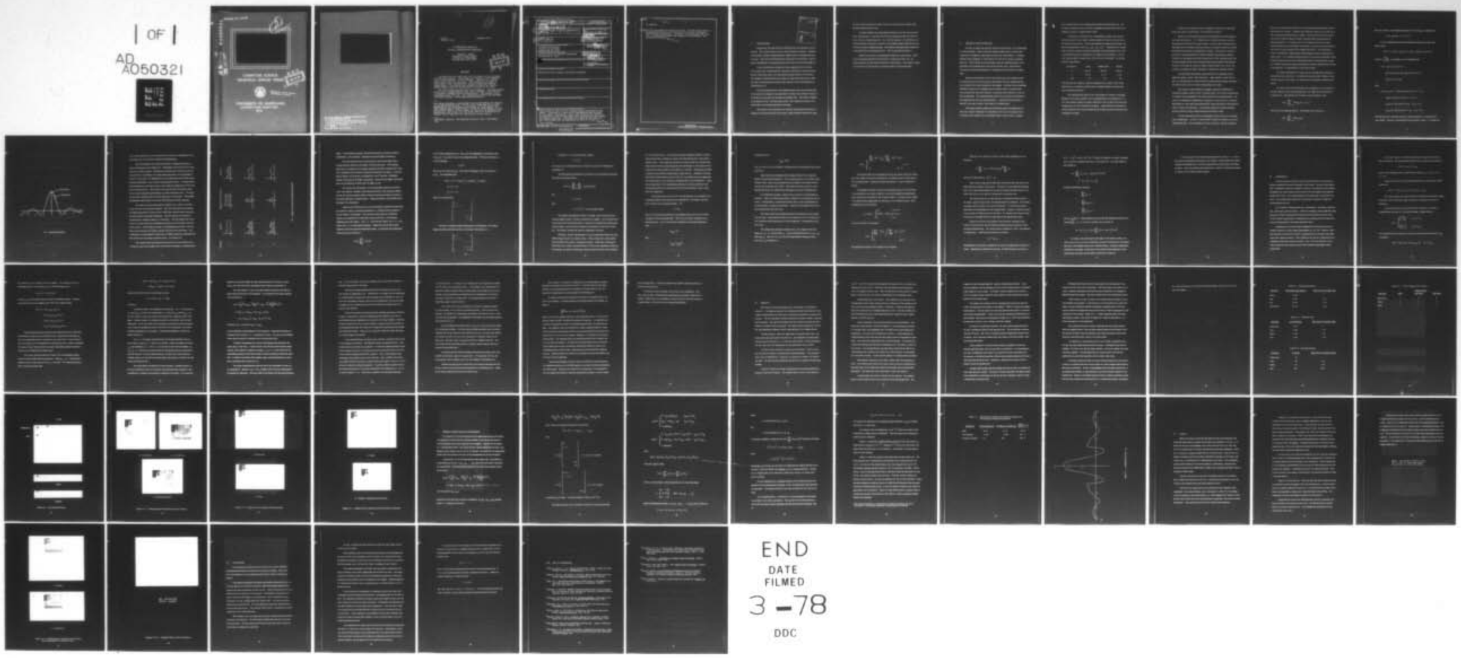
AD-A050 321 MARYLAND UNIV COLLEGE PARK COMPUTER SCIENCE CENTER F/G 5/8
A COMPARATIVE STUDY OF DIGITAL INTERPOLATION TECHNIQUES. (U)
NOV 77 E A HASER AFOSR-77-3271
UNCLASSIFIED TR-608 AFOSR-TR-78-0177 NL

| OF |
AD A050321



END
DATE
FILMED
3 -78
DDC

AFOSR-TR- 78-0177

AD A050321

AD No.
DDC FILE COPY

COMPUTER SCIENCE
TECHNICAL REPORT SERIES

D D C
FEB 23 1978
F

Approved for public release;
distribution unlimited.

UNIVERSITY OF MARYLAND
COLLEGE PARK, MARYLAND
20742

(2)

TR-608
AFOSR-77-3271

November 1977

A COMPARATIVE STUDY OF DIGITAL INTERPOLATION TECHNIQUES

Edward A. Haser*
Computer Science Center
University of Maryland
College Park, MD 20742

ABSTRACT

Interpolation has come into use in digital picture processing for many reasons. This paper is a report on the implementation and evaluation of four methods - replication, linear, cubic spline and sinc function. The evaluation is based on mean square error calculations, visual rating and computer usage costs. The results are limited but show that the best method depends upon the application for which it is used.

Two side investigations were also conducted. The mean square error of an interpolated picture can be reduced if the picture is optimally sampled before interpolation. The results of an experiment bear this out. An interpolation method was extended to color pictures. The method proved feasible.

The author gratefully acknowledges the encouragement and support of the Westinghouse Corp.; of Dr. Durga Panda and Prof. Azriel Rosenfeld of the University of Maryland; of Ralph Taylor and NASA Goddard Space Flight Center, which was instrumental in preparing this report; of Fran Fryer for her help in preparing the manuscript; and of the U.S. Air Force Office of Scientific Research, which supported the publication of the report.

---

*Permanent address: Westinghouse Electric Corp., Baltimore, MD.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFOSR-TR- 78-0177	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A COMPARATIVE STUDY OF DIGITAL INTERPOLATION TECHNIQUES.		5. TYPE OF REPORT & PERIOD COVERED Interim rept.,
		6. PERFORMING ORG. REPORT NUMBER TR-608
7. AUTHOR(s) Edward A. Haser		8. CONTRACT OR GRANT NUMBER(s) AFOSR-77-3271
9. PERFORMING ORGANIZATION NAME AND ADDRESS University of Maryland Computer Science Center College Park, MD 20742		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 2304/A2
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Office of Scientific Research/NM Bolling AFB, DC 20332		12. REPORT DATE Nov[redacted]77
		13. NUMBER OF PAGES 56
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Interpolation has come into use in digital picture processing for many reasons. This paper is a report on the implementation and evaluation of four methods - replication, linear, cubic spline and sinc function. The evaluation is based on mean square error calculations, visual rating and computer usage costs. The results are limited but show that the best method depends upon the application for which it is used.

Cont on reverse

last page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE UNCLASSIFIED

403018

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract

Two side investigations were also conducted. The mean square error of an interpolated picture can be reduced if the picture is optimally sampled before interpolation. The results of an experiment bear this out. An interpolation method was extended to color pictures. The method proved feasible.

I. INTRODUCTION

Interpolation has come into use in digital picture processing for several reasons. Data compression has been a major direction of research. Interpolation provides a means of regenerating the original picture, although with some error (8). Other uses of interpolation have appeared in the literature, such as picture enlargement (10) and generation of more data for better processing (11).

Various interpolation methods have been suggested in the literature. Four of these were implemented and compared on the basis of several criteria. Section II describes many of the interpolation methods found in the literature. The methods of implementation and results are reported in this section, along with some interesting observations by their authors, such as interpolation as convolution in (10).

Section III describes the actual implementation of the four methods used. It also lays out the design of the experiment to compare the methods in terms of the quality of the picture and computer processing costs. The quality is defined by an objective error, the mean square error, and a subjective measure, the visual effect of the interpolated picture on people.

The results of the experiment are tabulated and discussed in Section IV. Because of the small data base (one picture), generalizations could not be made, but some expected trends are noted, such as the fact that more complex interpolation yielded smaller errors.

In trying to decide what compression method to use in the mean squared error measurement, it was discovered that the sampling method can affect the error outcome of interpolation (8). As a side investigation, an experiment was performed which compared the results of the sampling method of Section III with a derived optimum sampling method. The optimum sampling method resulted in a noticeably better mean squared error. This is described in Section V.

Section VI performs one interpolation method on a color picture. Most of the processing reported in the literature is in black and white, but, the original impetus for this thesis arose from a color picture. The results of using interpolation on color pictures are much the same as in black and white.

II. REVIEW OF THE LITERATURE

In order to reduce the amount of storage of picture data, it is compressed by various methods. Some of the more elegant methods used to achieve this reduction are transform compression and predictive compression. A common method is point sampling, in which points to be stored are chosen at periodic intervals. This is known as comb filtering, since the sampling process can be described as the multiplication of the picture function with a comb function. The process of point compression is to represent a block of points by a single value.

Hummel and Rosenfeld (8) have done research into the problem of how to sample the data so that the picture reconstructed from the compressed data has a minimal error when compared to the original. They note that the sampling method for minimal error depends upon the method used for reconstruction. For reconstruction with linear splines, a piecewise linear function of a sinc (x) form is generated. The compressed data points are the result of convolution of the original picture with the derived function. A piecewise cubic function is derived in the case of using a cubic spline for reconstruction.

The authors found the difference between the optimal sampling and two other less complex samplings to be dependent on the rate of decrease of autocorrelation of the sampled (or compressed) points, which in turn is related to the autocorrelation of the original picture and the compression ratio. The two other methods were the comb filter sampling mentioned earlier and the unweighted average of a neighborhood of points.

If the picture is assumed to be a homogeneous random field, then the autocorrelation R of two random variables within the field is a function only of the distance between them. If the autocorrelation is assumed to decrease with distance exponentially, then $R(X_o, X_k) = R_K = R_o \lambda^k$. Using these two assumptions, which are common in picture processing, the mean squared error of the reconstructed picture as compared to the original can be predicted. For the case of linear spline reconstruction, the errors were calculated. An excerpt from the table follows:

Correlation	Comb	Neighborhood	Optimal
0	166.67	100.00	100.00
.5	22.53	17.18	14.61
.99	.34	.27	.22

This indicates that though the optimal weighting always yields a lower mean squared error, it does not always result in enough reduction to warrant the extra computation involved.

The expected mean square error was plotted as a function of sampling distance for the three methods, where sampling distance was defined as $-\ln \lambda$. For close samples optimal sampling reduced the error by about 35% from comb sampling and 17% from unweighted averaging. Larger distances increased the percent reduction from comb sampling but decreased the percent reduction from unweighted averaging.

Further details about this work, including the results of an experiment using cubic spline reconstruction, are presented in Section V.

Another use of interpolation in picture processing is for data expansion and visual enhancement. C.D. McGillem (10) at Purdue University has used interpolation on ERTS data for these reasons. He notes that the quantization of amplitude and the sampling at discrete spatial co-ordinates results in inevitable distortion when the data is reproduced for viewing. In attempts to enlarge the picture for visual examination the obvious method of repeating data samples resulted in a "graininess and artificial appearance" in the enlarged picture that was unacceptable and the author proposed using other methods of interpolation. Because he had found no criteria to indicate which method to use, three "conventional" methods were used - polynomial, trigonometric and sinc function.

The polynomial interpolator implemented was the Lagrange formula using four points to yield a cubic polynomial. When applied to actual picture data, the enlarged picture showed smooth transition between regions with differing contrast and a lack of overshoot and ghosting was evident.

The second method consisted of passing an nth order trigonometric polynomial through n points, which, the author states, is the same as generating the Fourier series expansion of the function defined by the sampled points. One reason for choosing the trigonometric interpolator is that errors are uniformly distributed over the region being interpolated, unlike polynomials which generate larger errors at the end points. Data enlarged using this method resulted in pictures similar to those obtained using the polynomial method.

It is not uncommon in picture processing to treat a picture as a sampled time varying signal. As such, it makes sense to talk of the Nyquist rate for the sampling function. If the sampling is at least at this rate, then the continuous signal can be regenerated exactly from the samples by convolving them with the appropriate sinc function. Problems noted with this method are the large set of data required and oscillations due to edges. The author notes that they may be overcome by using the discrete Fourier transform (DFT) and presented the following algorithm - 1) compute the DFT, 2) assume the data set is band limited and add zeros for the higher frequencies, 3) return to the spatial domain and scale the result upwards by the magnification factor. The interpolated values lie between the original points whose spacing has been increased by the added zeroes. The resulting enlarged pictures are of quality comparable to that obtained from the previous two interpolation schemes, though mention is made of a "mottled" appearance being characteristic of pictures generated by this type of interpolation. No explanation for this could be found.

The author concludes that a picture can be enlarged without altering its appearance in any major way. No method was chosen as best, though the comments in this and a later paper (11) indicate that the polynomial method is preferred.

The author notes that interpolation can be thought of as a convolution of the given function with an interpolating pulse, not unlike the third method described above. Interpolation can be written as (in the discrete case)

$$f(u) = \sum_{k=-\infty}^{\infty} f(k)\, g_k(u), \quad 0 \le u \le 1$$

where g_k is the weighting function. Convolution can be written as

$$f(u) = \sum_{k=-\infty}^{\infty} f(k)\, p(u-k).$$

Then there exists a correspondence between p (u-k) and g_k (u), specifically

$$p\ (u) = g_k\ (u+k) \qquad 0 \leq u + k \leq 1$$

For the Lagrange formula with equidistant samples and a cubic polynomial result

$$f\ (x) = L_{-1}\ f\ (x_{-1}) + L_0\ (x_0) + L_1\ (x_1) + L_2\ (x_2),\quad x_0 \leq x \leq x_1$$

Letting $u = \dfrac{X-X_0}{X_1-X_0}$, the weights L can be simplified and

$$f\ (u) = -\frac{1}{6} u\ (1-u)\ (2-u)\ f\ (-1)$$

$$+\frac{1}{2}(1+u)(1-u)\ (2-u)\ f\ (0) + \frac{1}{2}(1+u)\ u\ (2-u)\ f\ (1)$$

$$-\frac{1}{6}(1+u)\ u\ (1-u)\ f\ (2)$$

Then

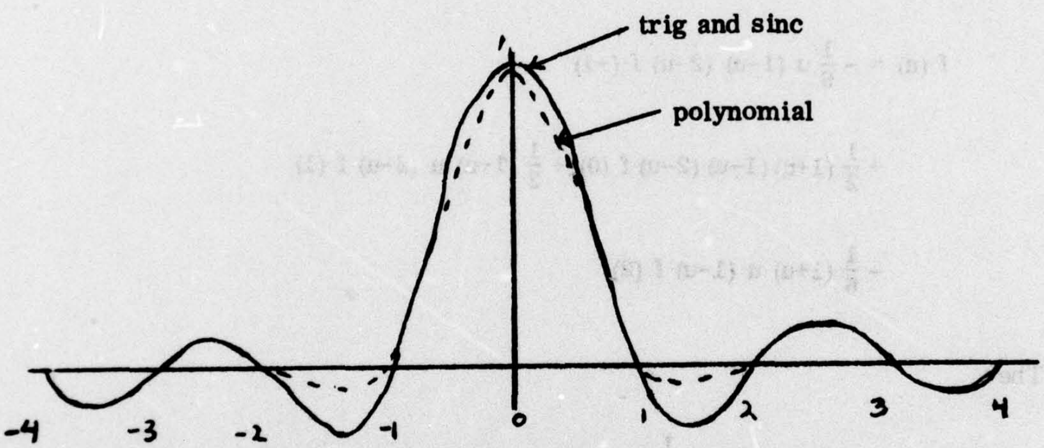
$$p\ (u) = g_{-1}\ (u-1) = -\frac{1}{6}(u-1)\ (2-u)\ (3-u),\quad 0 \leq u - 1 \leq 1$$

$$= g_0\ (u) = \frac{1}{2}(1+u)\ (1-u)\ (2-u) \qquad 0 \leq u \leq 1$$

$$= g_1\ (u+1) = \frac{1}{2}(1+u)\ (2-u)\ (1-u) \qquad 0 \leq u + 1 \leq 1$$

$$= g_2\ (u+2) = \frac{1}{6}(2+u)\ (3+u)\ (-1\ -u) \qquad 0 \leq u + 2 \leq 1$$

Plotting this gives a sinc-like function, shown in Figure II-1, along with the other pulses. The plot is reproduced from the authors' report. It is noted that



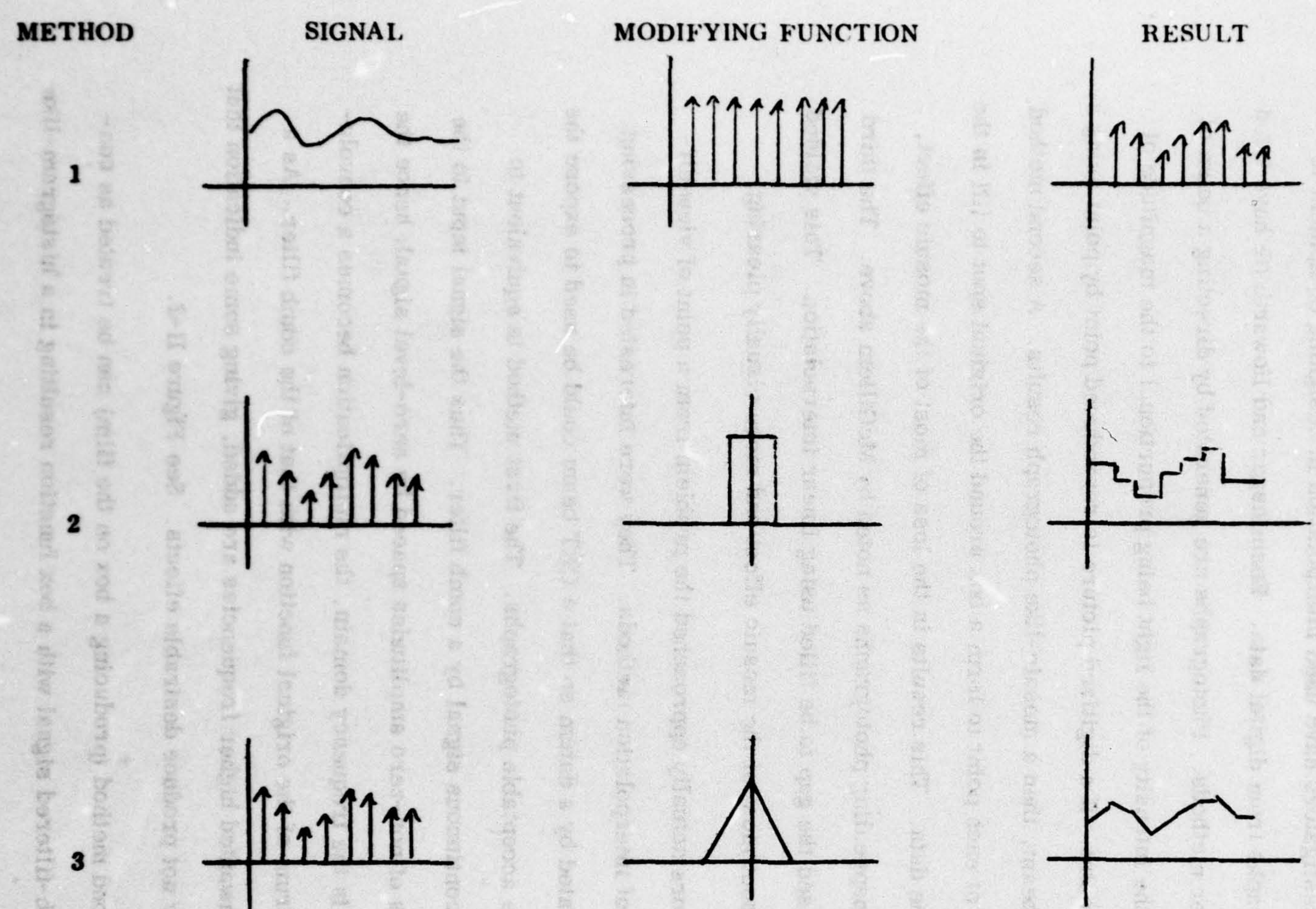
II-1. Interpolating pulses

these pulses approximate the sinc function form and that the trigonometric and sinc pulses are for all practical purposes indistinguishable.

Other investigators have used interpolation in designing equipment to generate photographs from digital data. Ensminsger and Howarth (6) have tried several low-order methods. Photographs are generated by directing a beam of light onto film, the intensity of the light being proportional to the magnitude of the data at each point. If a digitized picture is reproduced point by point using a moving focused beam, then a mosaic-like photograph results. A second method used replication of each point to form a box around the original spot to fill in the space between the data. This results in the loss of most of the mosaic effect, but in visually unappealing photographs as noted by McGillem above. The third method used caused the gap to be filled using linear interpolation. This yielded photographs without most of the mosaic effect and more visually pleasing.

The authors actually approached the problem from a point of view different than that of interpolation methods. They were interested in processing the signal generated by a datum so that a CRT beam could be used to expose the film and produce acceptable photographs. The first method is equivalent to multiplying the continuous signal by a comb filter. Thus the signal input to the beam is a series of non-zero amplitudes spaced by zero-level signal; hence the mosaic effect. In the frequency domain, the multiplication becomes a convolution of the spectrum of the original function with that of the comb filter. As a consequence, unwanted higher frequencies are added, giving some indication that this method may not produce desirable effects. See Figure II-2.

The second method (producing a box on the film) can be treated as convolving the comb-filtered signal with a box function resulting in a histogram-like



II-2. **Output of the three interpolation methods** signal. In the frequency domain, the spectrum derived in the first method is multiplied by a sinc function, reducing the unwanted higher frequencies.

The third method involves convolving the comb filtered signal with a triangle function (itself the convolution of two box functions). The resulting signal then is composed of the original sampled amplitudes connected with a signal of amplitude equal to linear interpolation between the two points. In the frequency domain, the spectrum is multiplied by a sinc^2 function, resulting in sharper attenuation of the higher frequencies. Because of this higher attenuation some of the sharpness of detail (such as edges) is lost.

The authors note that ideally, the incoming signal should be convolved with a sinc function, yielding a band-limited spectrum in the frequency domain, but the sinc function requires infinite samples and inherent negative values produce some difficulty in implementation. Using separability, the methods can be extended to two dimensions.

Andrews and Patterson (2) have also investigated methods of interpolation that result in visually acceptable photographs (or "psychovisually pleasing" to use the authors' terminology). They note that spline functions of different orders can be generated by convolving a square with itself, n convolutions yielding an nth order spline. For n = 1, a box function results, for n = 2, a triangle, for n = 4, a cubic spline (B-spline). These then will be used a basis functions which will yield the interpolated image. In particular the interpolated picture g(x) can be written as

$$g(x) = \sum_{i=1}^{N} c_i S_i(x) ,$$

for an N point original data set. The c_i are the weightings for the splines such that $g(x_i) = f(x_i)$ where $f(x_i)$ are the original samples. This can be written as a matrix equation

$$g = Ac$$

where g is the vector of $g(x_i)$, c the vector containing c_i and A the matrix of $S_i(x_i)$. For equidistant data,

$$S(x) = x_+^3 - 4(x-h)_+^3 + 6(x-2H)_+^3 - 4(x-3h)_+^3$$

$$x_+ = 0, \; x \leq 0$$

$$x_+ = x, \; x > 0$$

Then A is a band matrix:

$$\begin{bmatrix} 4 & 1 & & & & \\ 4 & 4 & 1 & & & \\ & & \cdot & & & \\ & & & \cdot & & \\ & & & \cdot & & \\ & & & 1 & 4 & 1 \\ & & & & 1 & 4 \end{bmatrix}$$

In order to completely define interpolation at the boundary, the authors assume the data is periodic and thus "circularize" the matrix A to

$$\begin{bmatrix} 4 & 1 & & & & 1 \\ 4 & 1 & 1 & & & \\ & & & \cdot & & \\ & & & \cdot & & \\ & & & 1 & 4 & 1 \\ 1 & & & & 1 & 4 \end{bmatrix}$$

Solving for c, A must be inverted, yielding

$$c = A^{-1} g$$

For large A, the circulant nature of A allows one to use FFT techniques in computing A^{-1}.

Two-dimensional functions such as pictures can use separable interpolation and the equations become

$$g(x, y) = \sum_{i=1}^{N} \sum_{j=1}^{N} C_{ij} S_i(x) S_j(y)$$

and

$$G = A\,C\,A$$

thus

$$C = A^{-1} G A^{-1} \quad \text{for the bicubic spline.}$$

The authors used splines of orders 1 through 4 and a piecewise linear function using the same A matrix to determine the weights. They conclude from their pictures that replication (order 1) results in an unsatisfactory appearance psychovisually, while the other methods all yield about the same effect on the eye. The bilinear method was easiest to implement, they note.

McGillem, Riemer and Mobasseri (11) have used interpolation with their ERTS scanner data for yet another reason. Their scanner has a point spread function (PSF) which causes a blurring of the data. Classically, knowing the PSF allows one to build a restoration filter to correct the degradation caused by the blurring, as long as the blurring is shift-invariant and thus can be modeled by a convolution process. The authors assumed a Gaussian model for the blurring and went about matching the model to the blurring aperture, from which a problem arose. They could only guarantee 3-5 points within the blurring width. They claim it has been found empirically that restoration is more easily accomplished when a large number of data points are available. Interpolation provided a solution to the problem of generating a larger data set. Choosing a Lagrange polynomial method based on earlier work (10), the scanner data was augmented. The scanner data was not originally sampled equally in both dimensions. Assuming the separability of the interpolation function, the authors generated more interpolation points in one dimension than the other producing a square output from the oblong input.

The authors make use of the fact that interpolation can be thought of as a convolution and that the blurring can be modelled as a convolution, and treat both as filters in the frequency domain. Let

$$G = HF$$

where F is the Fourier transform of the original picture and H is the Fourier transform of the blurring function. Then G is the Fourier transform of the blurred picture. Let I be the Fourier transform of the interpolating pulse. Then

$$G_{aug} = IG$$

and

$$F_{aug} = G_{aug}/H$$

Combining the two

$$F_{aug} = IG/H$$

and a new filter I/H can be defined, combining both the interpolation and restoration filters.

Experimentally, photographically enlarged pictures were compared against interpolated enlargements. Pure enlargement gave a very blurred and diffused result, while the interpolation process yielded a much better picture visually and maintained more detail. Some linear features were fuzzy in the interpolated image, but enhancement techniques were used to sharpen these.

As mentioned previously, pictures can be treated as two dimensional functions. Thus, any interpolation scheme available in two dimensions may be useful. Cartographers, in generating contour plots, require interpolation of two dimensional, irregularly spaced data. Shepard (14) mentions several functions and describes one of his own.

The author states that existing methods of interpolation can be divided into two types: single functions used over the entire data set or a collection of local functions joined appropriately at their boundaries. His method is of the latter type.

The interpolation function is defined over a local region R with data points $(x_i, y_i) = r_i$, having values z_i. Let the interpolated point be $p = (x_p, y_p)$ and value z_p. Also let $d(p, r_i)$ be the Cartesian distance between p and r_i. Then $f(p) = z_p$ is defined as:

$$
f(p) = \begin{cases} \sum_{i,R} (d_i)^{-u} z_i \Big/ \sum_{i,R} (d_i)^{-u}, & d_i \neq 0 \\ & , u > 0 \\ z_i & , d_i = 0 \end{cases}
$$

The function then is the weighted sum of the data points within the region, where the weight is inversely proportional to the distance from the data point to the interpolated point. Empirical results showed that u = 2 gave satisfactory results.

In defining the region R, some new constraints were put on the function. It was decided that the region should be a circle surrounding the interpolating point and should have between 4 and 10 points within it. An initial search radius C was defined and changed until the criterion of 4 to 10 points was met. New weighting factors were defined

$$
s_i = s(d_i) = \begin{cases} 1/d & 0 < d \leq c/3 \\ 27\,(d/c - 1)^2/4C & c/3 < d \leq c \\ 0 & c < d \end{cases}
$$

so that the function f (p) becomes

$$
f(p) = \begin{cases} \sum_{i,R} s_i^2 z_i \Big/ \sum_{i,R} s_i^2 & , d_i \neq 0 \\ z_i & , d_i = 0 \end{cases}
$$

No explanation is given for the ranges of the s function.

Direction can be taken into account; a directional weighting term was defined by

$$t_i = \sum_{j\,R} s_j \,(1 - \cos (r_i \, p \, r_j)) \Big/ \sum_{j\,R} s_j$$

and the s_i^2 replaced by $w_i = s_i^2 \,(1 + t_i)$.

Other factors could also be taken into account and slope and barrier considerations were added to the function. A barrier on a map could be something like a river which would indicate in the case of contours that data points on one side of the river have no effect on interpolation on the other side.

The author notes that the final function is continuously differentiable at all points, except barriers where the discontinuities are specified. The function also generates the original data points by definition. To avoid computational difficulties in interpolating near the original points, a region e around each point was set aside as having the value of the point. If e contains more than one point, all points are averaged and the average used for the interpolation value.

Splines have proven to be popular in the literature examined so far. Harder (7) notes that the aircraft industry developed surface splines for interpolating wing deflections. The surface spline is defined as a plate "that deforms in bending only." Deflections and loads are related by

$$D \nabla^4 W = q$$

The deflection of the spline is defined as the sum of the deflections of loads at N points. Integrating the differential equation, the deflection due to one point is

$W(i) = a + b_i^2 + (P/16\,\pi\,D)\,r^2 \ln r^2$, where the equation is in polar coordinate form, a and b are undetermined and P is the point load. The entire spline is then defined as

$$W = \sum_{i=1}^{N} (a_i + b_i \Gamma_i + (P_i/16\,\pi\,D)\,r^2 \ln r^2 ,$$

$$\pi_i^2 = (X - X_i)^2 + (y - y_i)^2$$

Using the equilibrium equations

$$\sum_i P_i = 0$$

$$\sum_i x_i P_i = 0$$

$$\sum_i y_i P_i = 0$$

and the fact that at large distances from the load the deflection should be zero specifies $\sum b_i = 0$, so that the equation can be written as

$$W = C_0 + C_1 x + C_2 y + \sum_i (P_i\ 16\ \pi/D)\ r_2^2 \ln r_i^2 .$$

To compare the performance of the splines with another method, the author chose to use a 21-term polynomial commonly implemented to interpolate data from a bell-shaped surface and a conical surface. Though no quantitative evaluation is presented, 3-axis plots of the results of both methods for both experiments were given and the spline clearly gave a better fit.

Two other papers were found using interpolation on data (3 , 4), but in both cases the information presented is not enough to clearly outline the method. Caprihan and Mendell use a recursive method based on spline theory developed by others. Bahr uses second order polynomials, based on a covariance function, to estimate errors found in ERTS imagery.

III. EXPERIMENT

From the previous section, it can be seen that there are several interpolation methods that produce acceptable visual results. But none of the papers offer a quantitative measure of quality or provide a cost analysis of the methods. Certainly if the visual results are the same for various methods, the cheapest method will be used. The implementation of the experiment is divided into two parts: 1) the interpolation methods to be used and 2) evaluation, taking into account quality and cost.

Four methods of interpolation were implemented - replication, line function, cubic spline and sinc pattern. A fifth was attempted, but the IBM SSP math package for Aitken-Langrange interpolation would not handle the picture data. All the methods implemented or attempted were reported upon in the literature. Each one implemented is discussed separately below.

Replication was the first method suggested in several papers and was roundly criticized as being visually unacceptable (2), (6), (10). However, intuition would say it would be the easiest to implement and no remarks were made about other quality measures. Thus replication was chosen to determine more quantitative information about the method. Also, it can be viewed as a zero-order polynomial and compared with the other methods using higher-order polynomials.

As the name implies, each point is repeated as required to obtain the desired expansion. The interpolation function in one variable can be written as

$$g(x) = x_i, \quad x_i \leq x < x_{i+1}$$

In the 4 times expansion used x would take the values x_i, $x_i + 0.25$, $x_i + 0.5$, $x_i + 0.75$.

Because pictures are two-dimensional entities the actual function implemented was:

$$g(x, y) = f(x_i, y_i), \quad x_i \leq x < x_{i+1}, \quad y_i \leq y < y_{i+1}$$

The linear function was chosen because (2) and (6) both indicated it yielded acceptable visual results and, again, intuitively, would seem very easy to implement.

Linear interpolation can be viewed as the evaluation of a series of weighted basis functions (6), the function being a triangle function:

$$t_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}} & x_{i-1} \leq x \leq x_i \\[2ex] \dfrac{x_{i+1} - x}{x_{i+1} - x_i} & x_i \leq x \leq x_{i+1} \end{cases}$$

The interpolated point would be the sum of two of these functions with the $f(x_j)$ as weights:

$$g(x) = f(x_i)\, t_i(x) + f(x_{i+1})\, t_{i+1}(x) \qquad x_i \leq x \leq x_{i+1}$$

The function t (x) is actually the linear spline. To account for the two-dimensional aspect, the function $z_{i\,j}$ (x, y) has the property that

$$z_{i\,j}(x, y) = z_i(x)\, z_j(y)$$

and thus $z_{i\,j}$ is a separable function, known as the bilinear spline. The interpolated point is then the weighted sum of four of the original points:

$$\begin{aligned} g(x, y) = {} & f(x_i, y_j)\, t_{i\,j}(x, y) + \\ & f(x_i, y_{j+1})\, t_{i\,j+1}(x, y) + \\ & f(x_{i+1}, y_j)\, t_{i+1\,j}(x, y) + \\ & f(x_{i+1}, y_{j+1})\, t_{i+1\,j+1}(x, y)\,. \end{aligned}$$

The separability property allowed much simplification in the implemention because it means one can interpolate in the x direction first and then use the semi-augmented matrix to interpolate in the y direction. In the actual program the columns were interpolated first and then the rows to assist in obtaining the required output format needed to generate photographs using the Dicomed machine at the Goddard Space Flight Center.

The x and y used were given the values of the corresponding original matrix row and column indices respectively. Thus $x_{i+1} - x_i = 1$ and the interpolated x took on values that were the $x_i \pm$ a fraction. The equations then simplify to the ones actually used:

$$g(x) = f(x_i)(x_{i+1} - x) + f(x_{i+1})(x - x_i)$$

$$= (f(x_{i+1}) - f(x_i))(x - x_i) + f(x_i)$$

which directly follows from the point slope formula

$$y - y_0 = m(x - x_0),\ m = \text{slope}$$

of algebra.

A problem arises at the boundaries of the picture. If $i = 1$, then what are x_{i-1} and $f(x_{i-1})$; if there are n points and $i = n$, what are x_{i+1} and $f(x_{i+1})$. In (2), they chose to assume the picture was periodic and "wrapped around" at the edges. Another scheme is to assume the picture is everywhere zero for undefined points. Since this author's first experience with pictures involved convolution in which the periodicity affects of the DFFT were undesirable, the second method of assuming zero was chosen. A third method might be to repeat the closest data point.

For $i = 1$, no problem existed because the interpolated points were always chosen as some $x_i + a$, where a is a fraction. Thus, no equation involving x_{i-1} was needed for $i = 1$. For $i = n$, a new point of value zero was added, i.e., $f(x_{n+1}) = 0$. This point was used strictly for calculations and never appeared in the final pictures. In actual implementation, the input matrix was placed in a working matrix, one row and one column larger than the input, the extra row and column containing zeroes.

The cubic spline was chosen for several reasons - personal interest, to do some preliminary work on the optimal sampling experiment (Chapter V) and availability of a suitable math package to implement the method. It is noted that splines have been the subject of much recent research (8) and that, in many cases, they will yield better interpolators than ordinary polynomials (5).

The cubic spline is a piece-wise third degree polynomial such that the spline and first derivative are continuous. The equation for the spline function over an interval is

$$s(x) = \frac{x - x_i}{h_i} f(x_{i+1}) + \frac{x_{i+1} - x}{h_i} f(x_i) - \frac{(x - x_i)(x_{i+1} - x)}{6 h_i} a$$

$$a = ((h_i + x - x_i) u_{i+1} + (h_i + x_{i+1} - x) u_i)$$

$$x_i \leq x \leq x_{i+1}, \; h_i = x_{i+1} - x_i, \; u_i = s''(x_i)$$

Note that $s(x_i) = f(x_i)$ and $s(x_{i+1}) = (f_{i+1})$.

A set of functions is then defined over each interval. Using the constraint of a continuous first derivative, n - 2 equations are found. If u_1 and u_n are assigned values then the system of equations can be written and solved.

The IMSL documentation (9) does not specifically state that this is the spline used, in this form. It does however state that the natural bicubic spline is used, where natural is defined as having $u_1 = u_n = 0$. It also states that the separability property of the bicubic spline is used to interpolate the points specified. It should be noted that other splines using a cubic polynomials are available, specifically the B-splines (2) (5).

The actual implementation made one call to the subroutine for each row of interpolation. Because $s(x_i) = f(x_i)$, original rows were also "interpolated" to simplify the algorithm. This also aided in the output of the Dicomed formatted tape. A row and column of zeros were added, just as in the linear method, to facilitate interpolation at the edges.

The fourth method implemented involved convolving the picture with a sinc function as suggested in (10). Theoretically, if a signal can be represented by a limited band of frequencies, then the signal can be sampled at a rate of at least twice the highest frequency with no loss of information, because the original signal can be reconstructed exactly by convolving the samples with an appropriate sinc function.

If the DFT is taken of the picture and the resulting spectrum is placed in a larger, zeroed matrix, then the resulting spectrum appears to be that of a band-limited function multiplied by the spectrum of the sinc function. Since the inverse transform will fill all entries possible, taking the inverse transform of the augmented spectrum will result in the original data points spaced by equidistant interpolated values. These are the values that would have been calculated by actually doing the convolution.

In the implementation, the picture was read into a complex array for use by the transform subroutine. The IBM SSP package provided the DFFT. Once transformed the spectrum was augmented. To keep the assumed periodic nature of the spectrum (13) intact, the spectrum had to be operated on first so that the zeroes could be added symmetrically around it. The q-shift operation takes advantage of the periodic nature of the spectrum. Once q-shifted, the zeroes were added around the spectrum, symmetrically with respect to the $(u, v) = (0, 0)$ element to achieve the desired expansion. A second q-shift was used to restore the augmented spectrum to the format required by the transform ($(u,v) = (0, 0)$ in matrix position $(1, 1)$). The inverse transform was then taken giving the interpolated result. A scaling of the resulting value was required; the scaling was I^2, where I is the expansion factor. For example, if the interpolation resulted in a picture with four times as many points as the original in any row or column, the interpolated picture would have to be multiplied by 16 to achieve the proper values for the original data. The implementation was written to handle only square matrices of data.

The result of the inverse transform is a matrix of complex numbers. To convert these to real values the following was done. If the real part was negative, the number was considered to be negative and was set to zero. If the real part was non-negative, the magnitude of the number was calculated and this was stored as the value.

All the methods were coded in such a way as to create line by line output for the Dicomed machine. The first three methods described can be viewed as weighted sums of non-negative basis functions and, thus, as long as the weights are positive (as they are in the case of pictures), no negative values can be generated. Because of this, no negativity check was added to the code. The last method could conceivably generate a negative number and the check was inserted as described above.

The spline and sinc function methods could generate numbers that were outside the allowable range for the grey scale. To compensate for this, any value greater than the highest grey level was changed to the highest level.

Because the input picture used 64 grey levels and the Dicomed used 256 levels, the data was rescaled before being output by multiplying it by 4. Using the 64 levels produced test pictures too dark to use.

The "quality" of a picture is a difficult item to ascertain since it usually depends on the purpose of the picture (12) (13). A picture used for visual presentation may contain distortions that is subjectively tolerated by the eye, but physically or quantitatively be unacceptable for processing (13).

In trying to quantitatively describe the quality of a processed picture relative to its original, a common measure is the mean squared error (MSE), defined as

$$\iint (f(x, y) - g(x, y))^2 \, dx \, dy$$

where f (x, y) is the original and g (x, y) the processed picture. It can be said then the lower the MSE, the better the quality of the picture. To adapt this as a means of measuring the quality of the interpolating methods, the following was done. The original picture was reduced by a simple sampled reduction, where the first and every fourth point in each row was kept and then the first and every fourth point in each of the new columns, yielding a 75% reduction in the number of data points. The reduced picture was then interpolated back to original size by each of the four methods. The MSE of the interpolated picture as compared to the original picture was calculated by squaring the difference between corresponding points in both pictures, summing the squares and finally taking the average. Double precision was used in the summing to prevent the possible loss of errors of small magnitude.

Because interpolation is used to "blow up" pictures for visual purposes, the regenerated pictures for the MSE measure were used to produce photographs for visual rating. Because of the small size of the pictures, it was decided to take the original and expand it using the interpolation methods to a more visible

size for comparison. A number of people were asked to order the pictures in terms of which was best.

To perform a cost evaluation, three factors were considered - CPU time, memory space required and prcgramming time needed to implement the method. Rather than try to synthesize a means to take all three into account in a single number, each was left to be compared individually.

IV. RESULTS

The results of the mean squared error calculations are tabulated in Table IV.1. As might be expected, the replication method, which uses the least amount of information from the picture to interpolate, produced the largest error measure. The line and spline functions produced much better results, relatively, than replication. Because these methods use more points to perform the interpolation, the better result is expected. The spline functions smooth fit over the data also apparently produces a better approximation to the original picture.

The sinc function, which one might expect to perform fairly well, was on par with replication and not the other two. An assumption was made about the spectrum of the picture, i.e., that it was band-limited. If this assumption was not valid, this is an explanation for the performance of the method.

In terms of time required to execute on the computer, Table IV.2 has the CPU time in minutes as reported by the operating system. The programs were run on an IBM 360/95. In general, the simpler the method, the shorter the time. The sinc function method took almost twice as long as the replication method.

Table IV.3 shows the storage requirements for the implementations in groups of 1024 bytes (K bytes). The original picture was 96 x 128 reduced to

24 x 32. The FFT required both dimensions to be powers of 2, so zeroes were added to fill out the array. This meant the sinc function method required a 128 x 128 complex matrix to run. This accounts for the large amount of storage listed. Other than that, the memory requirements were essentially equal.

Programming time is not shown. The methods for the most part were straightforward and a large amount of time was spent not on the method, but on how to use the math packages available. Therefore, it seemed that programming time would not be a meaningful indicator of cost. If all the methods had been written from scratch, then programming time would have been a useful measure, but they were not.

Visual evaluation of the pictures proved to be the most difficult. The small size of the pictures, as seen in Figure IV. 1, prevented good evaluation. To overcome this, the photographs were recorded on film and enlarge photographically. As can be seen in Figures IV. 2 and IV. 1, the photographic enlargement did not work well with data that did not present strong lines for good focus. The replication method did yield a sharp photograph. The people who produced the photographs offerred no explanation of the "fuzziness" of some of the pictures. The photographically enlarged spline method picture did not develop properly and, based on the results of the other pictures, it was decided not to be worth retrying. A color version (Figure VI. 1) did process acceptably.

Some general comments can be made, though, about the enlarged pictures. The mottled effect noted in (10) can be seen in the sinc method picture. The block effect of the replication method can be easily seen in the enlarged photograph. The unpleasant visual appearance is also very apparent.

A third attempt was made to visually rate the pictures. The original picture was recorded on film and enlarged four times photographically. The original was then enlarged digitally, using the interpolation methods. A four times enlargement was used because the size of the resulting picture was deemed large enough for easy viewing. To produce a picture using the sinc method, a 512 x 512 complex matrix would have been required, which exceeds the memory capacity of the machine used.

Five people were asked to rate the resulting three pictures in terms of clarity and sharpness with respect to the original. Table IV.4 shows the ratings of the pictures. The two people who chose the spline method picture as best are also amateur photographers. Those that rated the replication method picture as better agreed, when shown the photograph in Figure IV.2, that if the block effect had been that prevalent, their ratings would have changed.

In terms of a quantitative measure, the cubic spline method performed the best, yielding the smaller mean squared error. The line function, however, was only 7% worse. The other two methods produced significantly larger error measures, both in absolute measurement and relative to the best method, replication being 42% larger.

Time resulted in a wider deviation among the methods with the sinc method requiring almost twice as much time as replication. It is interesting to note that, excluding the sinc method, the results of the time comparison are the opposite of the MSE comparison with the spline method giving the worst time and replication giving the best. Apparently, reduced error comes at the expenses of more processing time.

Storage requirements rank the methods the same as time, but without the same wide spread of results. The extra 4 K bytes required by the spline method over replication is a small price to pay for the error reduction, since it is only a 4% increase in memory size.

Sharpness and clarity were the reasons given for choosing one of the pictures as best in the visual comparison. The visual effect of a picture is certainly a subjective phenomenon because the same properties were assigned to different pictures by people. The linear method was an all-around third choice.

These results all seem to support the statement that the quality of a picture depends on its purpose. If the picture is to be used strictly for visual purposes, then the replication method is apparently acceptable due to CPU time costs provided the enlargement factor is kept small (4 times enlargement in each dimension in this example, Figure IV.3). Larger expansion ratios will yield unacceptable results, such as that of Figure IV.2 which is equivalent to a 16 times enlargement in each dimension.

For results that must be visually acceptable and still closely approximate the original picture, then the spline method seems the best choice, if not the only one because of the line function's poor visual acceptance and the sinc function's and replication's poorer error performance.

To regenerate a compressed picture so as to obtain a minimum error, the line and cubic spline functions both have merit. Although the line function yielded a 7% worse error than the spline method, it ran in two-thirds of the time the spline required. If processing costs are a major factor, then the line method may be worth the increased error to reduce those costs.

It should be noted that these are rather limited results due to the lack of access to an adequate data base and not based on a large enough sample space to draw firm conclusions. In fact, to meaningfully relate the MSE measurement to an interpolation method, a large ensemble of pictures should be tested with the method (13). However, the results herein do seem to reflect the intuitive notion that the more complex the interpolation from a compressed picture, the greater the CPU processing costs in terms of time and storage, but the lower the error from the original.

Table IV. 1. Mean Squared Error

METHOD	MEAN SQUARED ERROR	RELATIVE TO LEAST MSE
Spline	18.62	1
Line	19.86	1.07
Sinc	24.70	1.33
Replication	26.36	1.42

Table IV. 2. Execution Time

METHOD	CPU MINUTES	RELATIVE TO LEAST TIME
Replication	0.083	1
Line	0.092	1.11
Spline	0.139	1.67
Sinc	0.162	1.95

Table IV. 3. Execution Storage

METHOD	K BYTES	RELATIVE TO LEAST SPACE
Replication	90	1
Line	92	1.02
Spline	94	1.04
Sinc	230	2.56

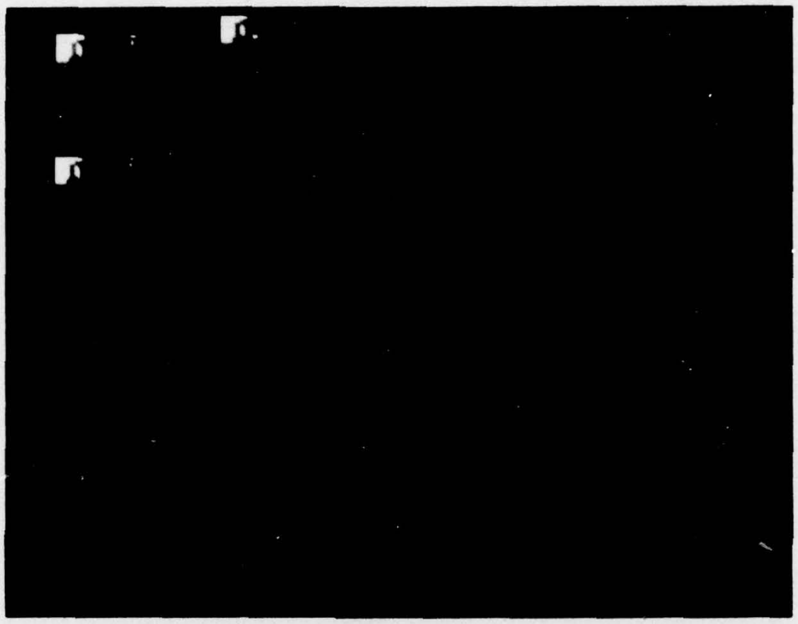
Table IV.4. Visual Ratings of the Pictures

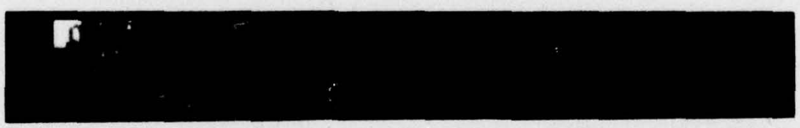
METHOD	TIMES RATED BEST	2ND BEST	3RD BEST
Replication	3	2	0
Linear	0	0	5
Spline	2	3	0

Linear

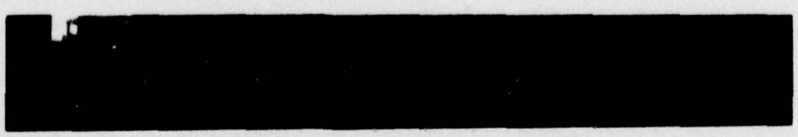
Replication

Sinc



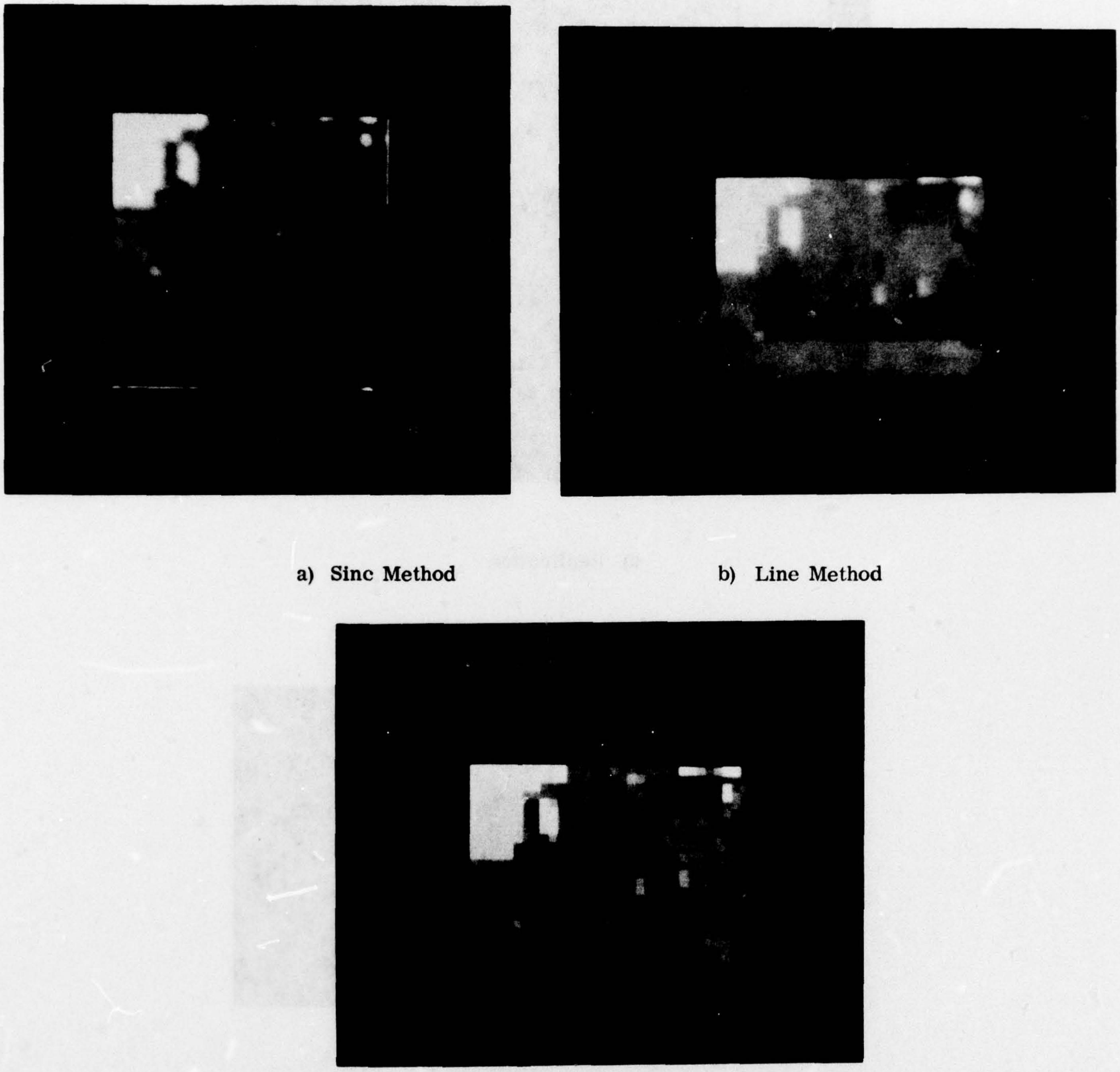


Spline



Original

Figure IV. 1. Interpolated Pictures

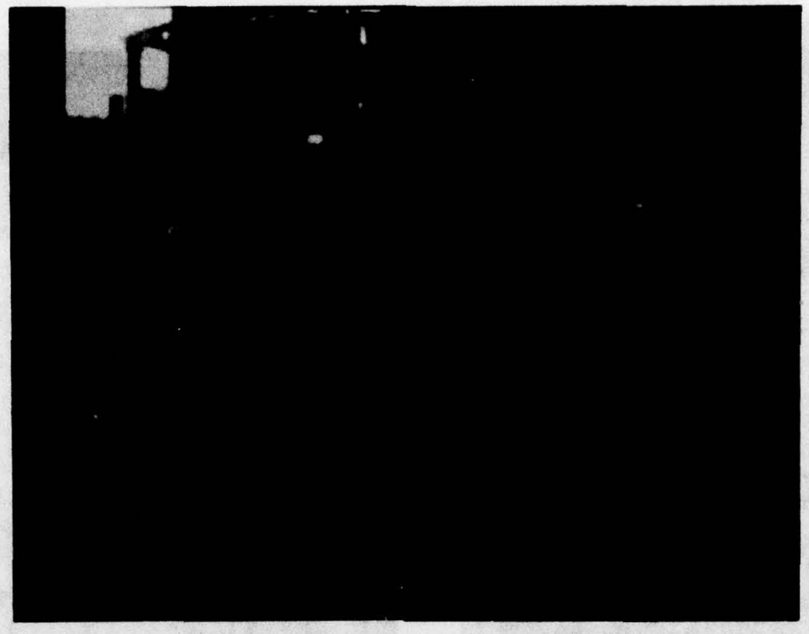


a) Sinc Method

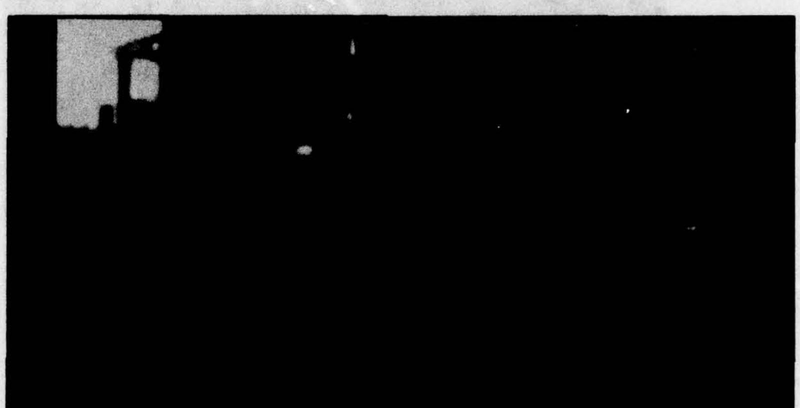
b) Line Method

c) Replication Method

Figure IV.2. Photographically Enlarged Interpolated Pictures

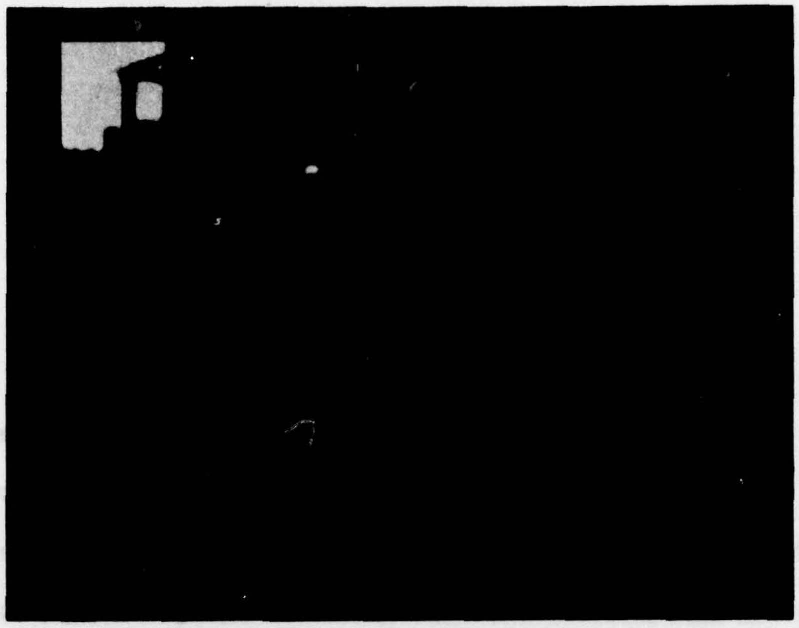


a) Replication



b) Line

Figure IV.3. Original Picture Enlarged with Interpolation



c) Spline

d) Original, Photographically Enlarged

Figure IV.3. Original Picture Enlarged with Interpolation (Continued)

V. MINIMAL-ERROR SAMPLING EXPERIMENT

If a picture is to be compressed and later regenerated using interpolation, one would want to chose the best methods possible so that the interpolated picture has minimal error when compared to the original. Hummel and Rosenfeld (8), as described earlier, have shown that for a given interpolation method, and using the mean square error as the error measure, the values for the compressed picture that will minimize the error in the interpolated one can be calculated.

In particular, for the one dimensional cubic spline case, the problem is to determine the vector $y = \{y_1, y_2, \ldots, y_n\}$ which will be stored to represent the original data. The interpolating function is the cubic spline defined earlier in this paper:

$$S_k(x) = \frac{x - x_k}{h_k} y_{k+1} + \frac{x_{k+1} - x}{h_k} y_k - \frac{(x - x_k)(x_{k+1} - x)}{6 h_k} a$$

$$a = ((h_k + x - x_k) u_{k+1} + (h_k + x_{k+1} - x) u_k), \; k = 1, 2, \ldots, n-1$$

over the interval (x_k, x_{k+1}).

Because the first derivatives must be continuous, $S'_k(x_k) = S'_{k-1}(x_k)$ and this yields n - 2 equations of the form

$$\frac{y_{k+1} - y_k}{h_k} - h_k \frac{2\,u_k + u_{k+1}}{6} = \frac{y_k - y_{k-1}}{h_k} + h_{k-1} \cdot \frac{u_{k-1} + 2\,u_k}{6}$$

which can be rearranged and written in matrix form

$$M\,u = D\,y, \quad u = \{u_2,\, u_3,\, \ldots,\, u_{n-1}\}$$

with

$$M = \frac{h}{6}\begin{bmatrix} 4 & 1 & & & & \\ 1 & 4 & 1 & & & \\ & & \cdot & & & \\ & & & \cdot & & \\ & & & & \cdot & \\ & & & & 1\,4\,1 & \\ & & & & 1\,4 & \end{bmatrix}, \text{ (n - 2) x (n - 2) matrix}$$

$$D = \frac{1}{h}\begin{bmatrix} 1 & -2 & 1 & & & & \\ & 1 & -2 & 1 & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & & & & 1\,-2 & 1 \end{bmatrix}, \text{ (n - 2) x n matrix}$$

provide all h_k are equal. It then is possible to write $u = M^{-1}\,D\,y$.

The spline function can be rewritten as the sum of two basis functions

$$p_k(x) = \begin{cases} (x - x_{k-1})/(x_k - x_{k-1}) & x_{k-1} \le x \le x_k \\ (x_{k+1} - x)/(x_{k+1} - x_k) & x_k \le x \le x_{k+1} \\ 0, \text{ otherwise} & \end{cases}$$

$$q_k(x) = \begin{cases} (x - x_{k-1})(x_k - x)(h + x - x_{k-1})/6h & x_{k-1} \le x \le x_k \\ (x - x_k)(x_{k+1} - x)(h + x_{k+1} - x)/6h & x_k \le x \le x_{k+1} \\ 0, \text{ otherwise} & \end{cases}$$

Then

$$S_k(x) = p_k(x)\, y_k + p_{k+1}(x)\, y_{k+1} - q_k(x)\, u_k - q_{k+1}(x)\, u_{k+1}$$

The entire spline is then

$$S(x) = \sum_j p_j(x)\, y_j - \sum_j q_j(x)\, u_j \; .$$

If the t_i are the points at which interpolation is to occur and using

$$C = \begin{bmatrix} 0 \ldots\ldots 0 \\ M^{-1} D \\ 0 \ldots\ldots 0 \end{bmatrix} \qquad \text{and } u = (u_1, u_2, \ldots, u_n)$$

then the interpolated values $s = (s(t_1), s(t_2), \ldots, s(t_m))$ can be written as

$$S = A\,y - A'\,C\,y = (A - A'\,C)\,y = J\,y$$

where

$$A \text{ is the matrix of } a_{ij} = p_j(t_i)$$

and

$$A' \text{ is the matrix of } a'_{ij} = q_j(t_i) .$$

A necessary condition to minimize the error $\sum_i (s(t_i) - f(t_i))^2$ yields the following

$$J^T J y = J^T f, \ f = (f(t_1), f(t_2), \ldots, f(t_m))$$

Then

$$y = (J^T J)^{-1} J^T f = B^{-1} J^T f$$

The values y for storage are the result of multiplying the original function by two matrices, which are known ~~and constitute~~ a set of weighting functions. If there are m original points to be reduced to n points then B is an n x n matrix and J an m x n matrix.

For two dimensions the weighting functions can be broken down into the product of two one-dimensional functions, if the two dimensional basis functions are separable. The authors note that in the case of bicubic splines, this is the case.

In the implementation, a reduction by 4 in both dimensions was needed, as was done in the earlier experiment. The x_k used for the basis functions p and q were the same as those resulting from the previous comb sampling, that is,

$$x_k = (k - 1) \times 4 + 1, \ k = 1, 2, \ldots, 33$$

To facilitate the algorithm for evaluating the basis functions, x_{33} was defined. Note that $h = 4$ in this case.

To account for the two dimensions, the $B^{-1} J^T$ matrix was used to first compress the original picture columnwise. Then the matrix was transposed to perform the row reduction.

Figure V.1 shows the weighting function derived for the 17th value of y_i, taken from the 17th row of $B^{-1} J^T$. Because $B^{-1} J^T$ was a 32 x 128 matrix, the figure shows the centermost of the functions. As expected, it has the general shape of a sinc function.

Table V.1 shows the results of this experiment and the earlier one. The mean squared error resulting from interpolating on the compressed data was 11.64. The error for the interpolation on the comb-sampled data was 18.63. The optimal sampling method resulted in a 37.5% reduction in the MSE. By the theory, optimal sampling should bring a reduction in MSE and apparently for this picture, the reduction holds true in practice. If similar results could be expected on other pictures, the extra processing cost may be well warranted.* Hummel and Rosenfeld note that the amount of reduction will decrease with increased correlation between sample points, so that the results of this one case cannot be generalized over all pictures. However it does indicate that if a picture must be compressed and later interpolated by cubic splines, optimal sampling methods should be investigated.

*The computer program as written did not attempt to optimize the use of matrices, so the storage requirement could probably be reduced.

Table V.1. Comparison of Results from Optimally-Sampled and Non-Optimally Sampled Experiments

MEASURE	COMB-SAMPLED	OPTIMALLY-SAMPLED	$\frac{\text{EXP 2}}{\text{EXP 1}} \times 100$
MSE	18.63	11.64	62.5%
CPU Minutes	0.139	0.355	255 %
Storage in K Bytes	94	218	232 %

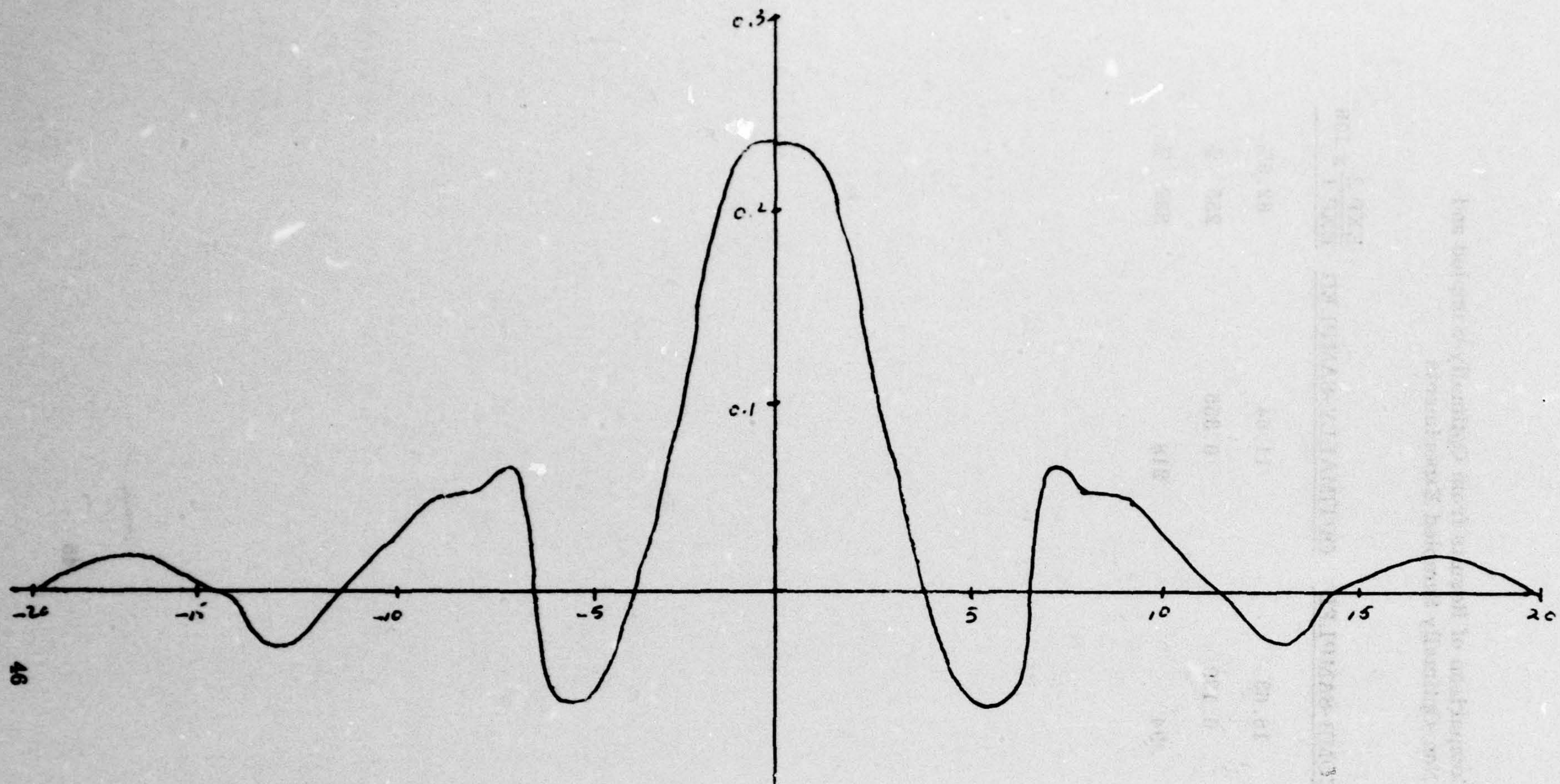


Figure V.1. Weighting Function, y_{17}

VI. COLOR

Most of the papers concerned with digital picture processing deal with black and white pictures, usually with the grey scale limited to 64 levels. It is known that the eye can distinguish more colors than grey levels (1) and, thus, the use of pseudo-color for picture enhancement has become widespread. Moreover, pictures are being processed that are in natural color. Both the Jet Propulsion Laboratory and the University of Southern California (1) have done work in this area, as witnessed by the Viking pictures from Mars. Yet no published papers could be found in the literature despite a professional, computerized literature search by the Goddard Space Flight Center through government reports and general publications.

Because color pictures exist and color display devices are becoming better technically and lower in cost (15), it would seem appropriate to see if the results of processing in black and white extend into color.

Color pictures usually exist as the composite of three separate color components, red, green and blue, just as the guns of a color TV are arranged. Color processing is mentioned briefly in (1), which suggests that because a color picture can be broken into three monochromatic components, that each be treated individually. This would also tend to be the common-sense approach.

Looking at the mechanics of grey scales, at least with the Dicomed machine used in this project, gives a possible explanation of why the color components can be treated separately. On the Dicomed, the grey levels are white light at various intensities, black being the lowest intensity and white being the maximum intensity. Processing, then, affects the intensity of the white light to produce the new greys. White can be generated by displaying equal amounts of red, blue and green, greys being produced by varying the intensity of each color to the same degree. It follows that as long as each color is processed the same, they can be operated on separately and recombined, producing the same result as with just using grey levels.

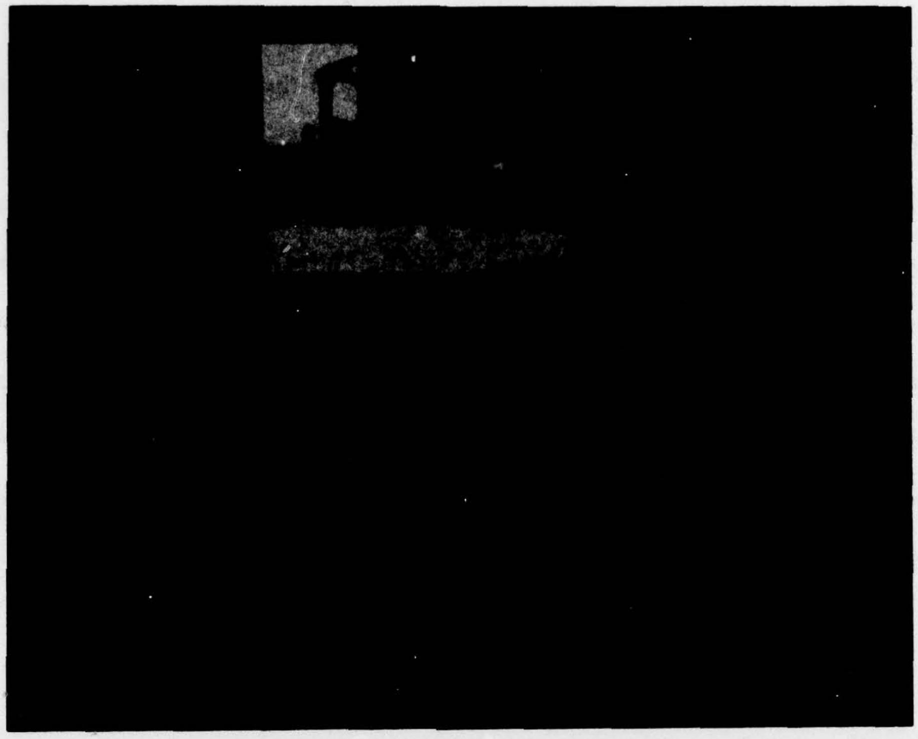
As can be seen via various color diagrams (12) (15), hues are continuous, that is, there are no discontinuities in the color components that cause the hue to change radically with small changes in the components. Hence interpolation of each component should produce a picture with no visible departures from the colors of the originals. Transitions from hue to hue should be smooth. Also, the effects seen by processing a monochromatic representation should be visible in the color picture.

Figure VI-1 bears this out. The black and white picture used previously was actually the green component of the color house picture. All three components were sample reduced by a factor of 4, as was done in the previous experiments, and expanded to original size using cubic spline interpolation. The components were then recombined to produce the final picture.

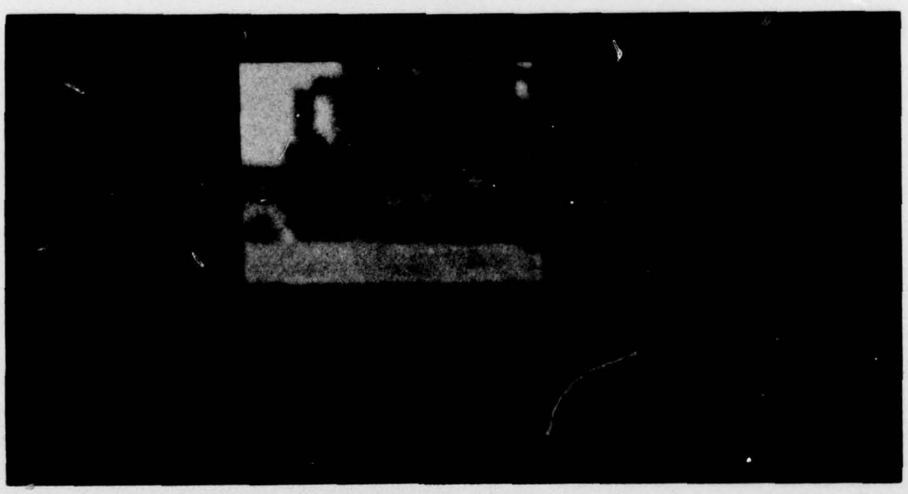
Comparing the result of the monochromatic and color processing, the general features of the objects are the same. The distorted porch roof and pillar are present in both pictures. The sampling and interpolation in both cases had the same effect.

Comparing the original color picture and the interpolated one, the hue throughout both pictures appear to be the same, as do the intensities of the colors. There are no visible departures of hue in the regenerated picture from the original that can be perceived. Unfortunately, as mentioned previously, the photographic enlarging process could not be done well with the interpolated pictures. The film used was Vericolor II, not a film most laboratories work with regularly. In verification of the above statements, Figure VI-2 has a Polaroid photograph, at the original size, showing the interpolated picture at top and the original underneath it.

<u>Note</u>: The color pictures (Figs. VI - 1 and 2) have been reproduced here in black and white.

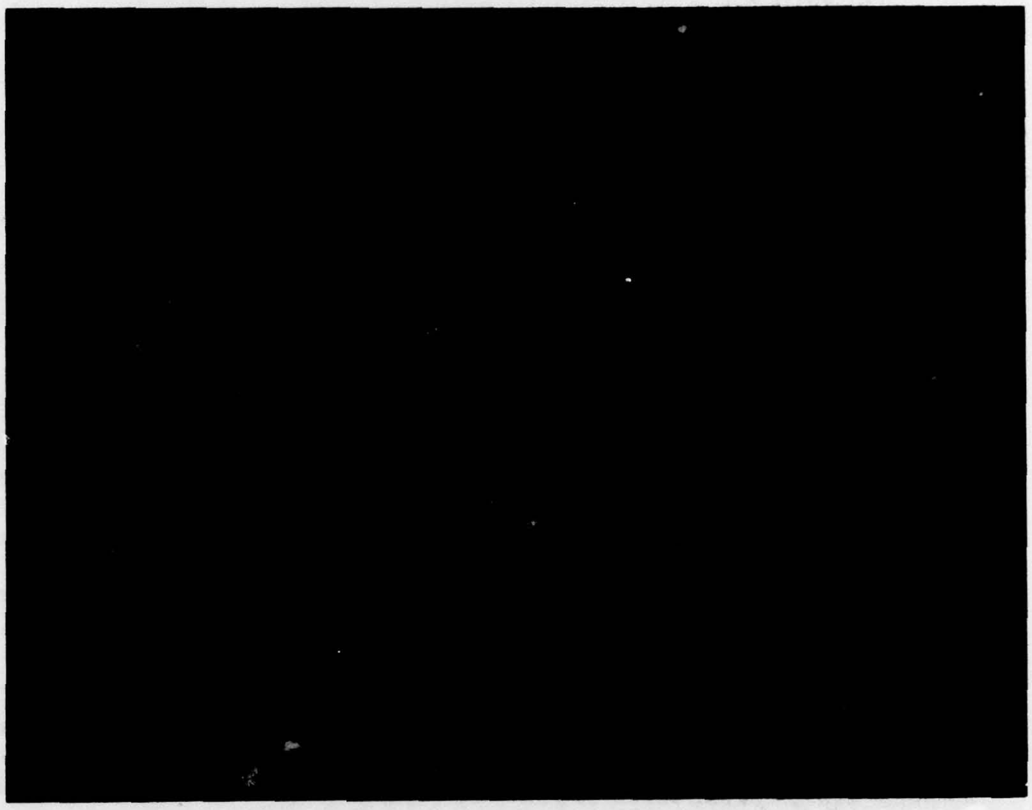


a) Original



b) Interpolated

Figure VI-1. Photographically Enlarged Color Pictures (both photographically enlarged 4 times)



top - interpolated
bottom - original

Figure VI-2. Original Size Color Pictures

VII. CONCLUSIONS

The preceding sections have shown that there are several methods of interpolation described in the literature on picture processing. Four of the more interesting ones were implemented and tested in order to compare the results.

Cubic spline interpolation and linear interpolation worked well as far as the mean square error test was concerned, while interpolation using the sinc function and replication did not perform as well. Visual comparison proved very difficult due to the small size of the pictures. Photographic enlargement was used to "blow up" the original to a viewable size. Due to computer memory limitations, the sinc method could not be used for this. The linear method resulted in the poorest picture. The cubic spline and replication methods essentially tied for best picture. The expected "block" effect of replication was small enough not to be a visual annoyance.

With regard to cost, the replication and linear methods used the smallest amount of time and space. The cubic spline method used about 67% more time than replication. The sinc function used twice as much time and 2.5 times as much memory compared to replication.

Overall, the MSE and visual tests seem to give the cubic spline method an edge over the others.

The limited data base of one picture did not provide a broad enough sample space to draw firm conclusions, but the results of the experiment fit those reported in the literature and in the case of the MSE measurement, the smoother the interpolating curve, the better the result, excluding the sinc function.

Two side investigations were made, one into optimal compression sampling to minimize error after regeneration and the other into color. The theory of optimal sampling provided a means for deriving the samples that when interpolated will yield minimal error as compared to the original. Implementing this for reducing four points to one in each dimension, the results showed a 37% reduction in error.

Color proved to be manageable by breaking the picture into three color components and interpolating each separately, recombining them for the final result. The experiment produced no unusual results and yielded a picture with the same shapes as the black and white processing. Photographic enlargements had the same problem as the black and white enlargements. The color also varied the developed pictures having different coloration than the Polaroid pictures of the same data. Color in general is more difficult to work with according to the people who make the prints and is subject to much variation unless a very controlled environment exists.

Two questions have arisen from this work that have been left unanswered. The first is: if the error can be reduced for cubic spline interpolation, would the other methods using the same compressed data also yield reduced errors? This would imply that instead of deriving the weighting functions for each interpolation method, only one general set of functions need be derived.

The second is that if the samples to be saved in optimal sampling can be written as a convolution of a weighting function with the original data as in (8), and interpolation can be treated as a convolution as in (10), then the following should be true.

$$g = p * r * f$$

where r is the optimal sampling function and p is the interpolating pulse. If $f = g$, then the interpolation has exactly reproduced the picture. Taking the Fourier transform, it follows then that

$$F = P R F$$

and, thus, that $P(u, v) R(u, v) = 1$ for all u, v. If the interpolating method and pulse are known, then the optimal sampling method should be determined.

VIII LIST OF REFERENCES

[1] Andrews, Harry C., A.G. Tescher and Richard P. Kruger, Image processing by digital computer, IEEE Spectrum 9, 1972, 20-32.

[2] Andrews, Harry C. and Claude L. Patterson, Digital interpolation of discrete images, IEEE Transactions on Computers C-25, 1976, 196-202.

[3] Bahr, H.P., Interpolation and filtering of ERTS imagery, Proceedings of the International Society of Photogrammetry Symposium, Stuttgart, Sept. 2-6, 1974, 235-243.

[4] Caprihan, A. and W.W. Mendell, Resolution improvement of remote sensing data, Proceedings of Hawaii International Conference on System Sciences, Honolulu, June 9-11, 1973, 159-196.

[5] Dahlquist, Gremund and Ake Bjorck, Numerical Methods, Translated by Ned Anderson, Prenctice Hall, Englewood Cliffs, New Jersey, 1974.

[6] Ensminger, R.L. and R.F. Howarth, An inner raster CRT photo generator, Proceedings of the SID, 11, 1970, 165-175.

[7] Harder, Robert L. and Robert N. Desmaricas, Interpolation using surface splines, Journal of Aircraft 9, 1972, 189-191.

[8] Hummel, Robert A. and A. Rosenfeld, Minimal error sampling, Technical Report 380, University of Maryland Computer Science Center, 1975.

[9] International Mathematics and Statistics Libraries Inc., Library 1 Reference Manual, edition 5, Houston, 1975.

[10] McGillem, C.D., Interpolation of ERTS-1 multispectral scanner data, LARS Information Note 22175, Purdue University Laboratory for Applications of Remote Sensing, 1975.

[11]McGillem, C.D., T.E. Riemer and G. Mobasseri, Resolution enhancement of ERTS imagery. Proceedings of the Symposium on Machine Processing of Remotely Sensed Data, West Lafayette, Indiana, June 3-5, 1975, 3A21-3A29.

[12]Moik, Johannes G., Introduction to Computer Image Processing, Goddard Space Flight Center, 1973.

[13]Rosenfeld, Azriel and Avinash C. Kak, Digital Picture Processing, Academic Press, New York, 1976.

[14]Shepard, Donald, A two dimensional interpolation function for computer mapping of irregularly spaced data, Paper 15, Harvard University Laboratory for Computer Graphics and Spatial Analysis, 1968.

[15]Wells, Donald C., Interactive image analysis for astronomers, Computer 10, 1977, 30-34.